IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of:	)	
	)	
Inventor: David E. Simmen	)	Examiner: Cindy Nguyen
	)	
Serial #: 10/807,871	)	Group Art Unit: 2161
	)	
Filed: March 24, 2004	)	Appeal No.: _____
	)	
Title: QUERY OPTIMIZATION TECHNIQUE FOR OBTAINING IMPROVED CARDINALITY ESTIMATES USING STATISTICS ON PRE-DEFINED QUERIES	)	

BRIEF OF APPELLANT

MAIL STOP APPEAL BRIEF - PATENTS

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

In accordance with 37 CFR §41.37, Appellant's attorney hereby submits the Brief of Appellant on appeal from the final rejection in the above-identified application, as set forth in the Office Action dated August 17, 2010.

Please charge the amount of $540 to cover the required fee for filing this Appeal Brief as set forth under 37 CFR §41.37(a)(2) and 37 CFR §41.20(b)(2) to Deposit Account No. 09-0460 of IBM Corporation, the assignee of the present application.

Also, please charge any additional fees or credit any overpayments to Deposit Account No. 09-0460 of IBM Corporation.

I. REAL PARTY IN INTEREST

The real party in interest is International Business Machines Corporation, the assignee of the present application.

II. RELATED APPEALS AND INTERFERENCES

There was a related appeal filed in U.S. Patent Application Serial No. 09/669,556, which is the parent of this application. A Board Decision was rendered on July 19, 2010. A copy of that Board Decision is identified in the Related Proceedings Appendix and attached hereto.

There was a related appeal filed in U.S. Patent No. 7,080,062. A Board Decision was rendered on May 20, 2005. A copy of that Board Decision is identified in the Related Proceedings Appendix and attached hereto.

Appeals were also filed in related U.S. Patent Nos. 6,738,755, 6,847,962 and 7,080,062, but these patents were allowed after the filing of Appeal Briefs.

III. STATUS OF CLAIMS

Claims 1-33 are pending in the application.

Claims 3, 14 and 25 were objected to as being dependent upon a rejected base claim, but were indicated as being allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Claims 4-11, 15-22 and 26-33 were also objected to because they depend on claims 3, 14 and 25, but otherwise were allowable.

Claims 1-2, 12-13 and 23-24 were rejected under 35 U.S.C. §102(e) as being anticipated by U.S. Patent No. 6,363,371 (Chaudhuri).

Claims 1-2, 12-13 and 23-24 are being appealed.

IV. STATUS OF AMENDMENTS

No response was submitted subsequent to the final Office Action dated August 17, 2010..

V. SUMMARY OF CLAIMED SUBJECT MATTER

Appellant's invention, as recited in independent claims 1, 12 and 23, are generally directed to optimizing execution of a query that accesses data stored on a data store connected to a computer.

Independent claim 1 recites a method of optimizing execution of a query that accesses data stored on a data store (104) connected to a computer (102). (See page 3, lines 19-29; page 4, lines 25-27 referring to 102 in FIG. 1; page 5, lines 13-28 referring to 102-108 and 114-124 in

FIG. 1.) The method includes the step of using statistics on one or more expressions of one or more pre-defined queries to determine an optimal query execution plan for the query. (See page 3, lines 19-29; page 6, lines 18-22 referring to 124 in FIG. 1; page 14, line 19 – page 18, line 7, referring to 124 in FIG. 1; page 18, line 9 – page 29, line 30, referring to 124 in FIG. 1; and page 30, line 2 - page 31, line 20 referring 200-204, 300-310, and 400-416 in FIGS. 2, 3, 4A and 4B, respectively.) In addition, the method includes the step of executing the optimal query execution plan for the query in order to access the data stored on the data store (104) connected to a computer (104) and then output the accessed data. (See page 3, lines 19-29; page 4, lines 25-27 referring to 102 in FIG. 1; page 5, lines 13-28 referring to 102-108 and 114-124 in FIG. 1.)

Independent claim 12 recites an apparatus for optimizing execution of a query. The apparatus includes a computer (102) having a data store (104) coupled thereto, wherein the data store (104) stores data. (See page 3, lines 19-29; page 4, lines 25-27 referring to 102 in FIG. 1; page 5, lines 13-28 referring to 102-108 and 114-124 in FIG. 1.) In addition, the apparatus includes one or more computer programs (108-124), performed by the computer (102), for using statistics on one or more expressions of one or more pre-defined queries to determine an optimal query execution plan for the query, and executing the optimal query execution plan for the query in order to access the data stored on the data store (104) connected to a computer (102) and then output the accessed data. (See page 3, lines 19-29; page 4, lines 25-27 referring to 102 in FIG. 1; page 5, lines 13-28 referring to 102-108 and 114-124 in FIG. 1; page 6, lines 18-22 referring to 124 in FIG. 1; page 14, line 19 – page 18, line 7, referring to 124 in FIG. 1; page 18, line 9 – page 29, line 30, referring to 124 in FIG. 1; and page 30, line 2 - page 31, line 20 referring 200-204, 300-310, and 400-416 in FIGS. 2, 3, 4A and 4B, respectively.)

Independent claim 23 recites an article of manufacture comprising a program storage medium readable by a computer (102) and embodying one or more instructions executable by the computer (102) to optimizing execution of a query that accesses data stored on a data store (104) connected to the computer (102). (See page 3, lines 19-29; page 4, lines 25-27 referring to 102 in FIG. 1; page 5, lines 13-28 referring to 102-108 and 114-124 in FIG. 1.) The instructions include using statistics on one or more expressions of one or more pre-defined queries to determine an optimal query execution plan for the query. (See page 3, lines 19-29; page 6, lines 18-22 referring to 124 in FIG. 1; page 14, line 19 – page 18, line 7, referring to 124 in FIG. 1; page 18, line 9 – page 29, line 30, referring to 124 in FIG. 1; and page 30, line 2 - page 31, line

20 referring 200-204, 300-310, and 400-416 in FIGS. 2, 3, 4A and 4B, respectively.) In addition, the instructions include executing the optimal query execution plan for the query in order to access the data stored on the data store connected to a computer and then output the accessed data. (See page 3, lines 19-29; page 4, lines 25-27 referring to 102 in FIG. 1; page 5, lines 13-28 referring to 102-108 and 114-124 in FIG. 1.)

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

1. Whether claims 1-2, 12-13 and 23-24 are anticipated under 35 U.S.C. §102(e) by U.S. Patent No. 6,363,371 (Chaudhuri).

VII. ARGUMENTS

A. Arguments directed to the first grounds for rejection: Whether claims 1-2, 12-13 and 23-24 are anticipated under 35 U.S.C. §102(e) by U.S. Patent No. 6,363,371 (Chaudhuri).

1. Claims 1, 12 and 23

i. The Office Action Rejections

On page (3) of the Office Action, claims 1-2, 12-13 and 23-24 were rejected under 35 U.S.C. §102(e) as being anticipated by Chaudhuri et al., U.S. Patent No. 6,363,371 (Chaudhuri)..

ii. Appellant's Claimed Invention

Appellant's claimed invention, as recited in independent claims 1, 12 and 23, is directed to optimizing execution of a query that accesses data stored on a data store connected to a computer. Claim 1 is representative and recites the steps of using statistics on one or more expressions of one or more pre-defined queries to determine an optimal query execution plan for the query, and executing the optimal query execution plan for the query in order to access the data stored on the data store connected to a computer and then output the accessed data.

iii. The Chaudhuri Reference

Chaudhuri describes an essential statistics identification utility tool that attempts to reduce or minimize the overhead associated with statistics by identifying from an initial set of statistics a set of essential statistics that provide a query optimizer with the ability to choose

among query execution plans with minimized loss in accuracy as compared to using the initial set of statistics. The set of essential statistics is identified as a subset of the initial set of statistics that is equivalent to the initial set of statistics with respect to each query of a workload. The subset of statistics is equivalent to the initial set of statistics if an execution plan for each query using the subset of statistics is the same as an execution plan for that query using the initial set of statistics and/or if a cost estimate to execute each query against the database using the subset of statistics is within a predetermined amount of a cost estimate to execute that query against the database using the initial set of statistics. The subset of statistics may be identified such that any proper subset of the subset of statistics is not equivalent to the initial set of statistics with respect to each query. The subset of statistics may also be identified such that an update cost or size for the subset of statistics is minimized.

iv. Appellant's Claimed Invention Is Patentable Over The Cited Reference

Appellant's claimed invention is patentable over the cited reference, because it includes a combination of limitations not taught or suggested by the Chaudhuri reference. Specifically, the reference does not teach or suggest the steps or elements of the independent claims comprising "using statistics on one or more expressions of one or more pre-defined queries to determine an optimal query execution plan for the query."

Nonetheless, Chaudhuri is cited by the Office Action as teaching all of the steps or elements of the independent claims 1, 12 and 23.

The portions of Chaudhuri cited by the Office Action are set forth below:

> Chaudhuri: Col. 2, lines 14-15
> A method identifies statistics for use in executing one or more queries against a database. The method may be implemented by computer-executable instructions of a computer readable medium. A database system may perform the method with suitable means.
>
> Chaudhuri: Col. 2, line 59 to col. 3, line 12
> For each statistic of the initial set of statistics, a respective set of queries may be identified from a workload of queries such that that statistic is potentially relevant to each query in the respective query set and such that each query in the respective query set has estimated execution costs greater than any other potentially relevant query of the workload. For each statistic of the initial set of

statistics, whether the initial set of statistics without that statistic is equivalent to the initial set of statistics with respect to each query in the respective query set may then be determined, and, if not, that statistic is included in a first subset of statistics. The one or more queries may then be identified from the workload as each query of the workload such that the first subset of statistics is not equivalent to the initial set of statistics with respect to that query.

The subset of statistics may be identified by identifying a subset of the initial set of statistics, determining whether such an identified subset of statistics is equivalent to the initial set of statistics with respect to each query, and repeating these steps for other subsets of the initial set of statistics. These steps may be repeated until an identified subset of statistics is equivalent to the initial set of statistics with respect to each query. Subsets of the initial set of statistics may be identified in increasing order of update cost or size.

Chaudhuri: Col. 4, lines 57-62

With reference to FIG. 1, an exemplary system for implementing the invention includes a general purpose computing device in the form of a conventional personal computer 120, including a processing unit 121, a system memory 122, and a system bus 123 that couples various system components including system memory 122 to processing unit 121. System bus 123 may be any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using any of a variety of bus architectures. System memory 122 includes read only memory (ROM) 124 and random access memory (RAM) 125. A basic input/output system (BIOS) 126, containing the basic routines that help to transfer information between elements within personal computer 120, such as during start-up, is stored in ROM 124. Personal computer 120 further includes a hard disk drive 127 for reading from and writing to a hard disk, a magnetic disk drive 128 for reading from or writing to a removable magnetic disk 129, and an optical disk drive 130 for reading from or writing to a removable optical disk 131 such as a CD ROM or other optical media. Hard disk drive 127, magnetic disk drive 128, and optical disk drive 130 are connected to system bus 123 by a hard disk drive interface 132, a magnetic disk drive interface 133, and an optical drive interface 134, respectively. The drives and their associated computer-readable media provide nonvolatile storage of computer-readable instructions, data structures, program modules and other data for personal computer 120. Although the exemplary environment described herein employs a hard disk, a removable magnetic disk 129 and a removable optical disk 131, it should be appreciated by those skilled in the art that other types of computer-readable media which can store data that is accessible by a computer, such as magnetic cassettes, flash memory cards, digital video disks, Bernoulli cartridges, random access memories (RAMs), read only memories (ROMs), and the like, may also be used in the exemplary operating environment.

Chaudhuri: Col. 6, lines 25-27

Database server 220 processes queries, for example, to retrieve, insert, delete, and/or update data in database 210. Database system 200 may support any

suitable query language, such as Structured Query Language (SQL) for example, to define the queries that may be processed by database server 220. Suitable SQL queries include, for example, Select, Insert, Delete, and Update statements. Database server 220 for one embodiment comprises the Microsoft.RTM. SQL Server.

Chaudhuri: Col. 6, lines 30-40

Database server 220 comprises a storage engine 222 for accessing data in database 210. To enhance performance in processing queries, database server 220 uses indexes to help access data in database 210 more efficiently. An index may be single-column or multi-column and may be clustered or non-clustered. Database server 220 comprises a query optimizer 224 to generate efficient execution plans for queries with respect to a set of indexes. **In generating execution plans, query optimizer 224 relies on statistics on column(s) of table(s) referenced in a query to estimate, for example, the cost in time to execute the query against database 210 using more than one possible execution plan for the query. Query optimizer 224 may then choose among possible execution plans for the query.** The notations Plan(Q,S) and Cost(Q,S) respectively represent the plan chosen by query optimizer 224 for a query Q and the execution cost of query Q estimated by query optimizer 224 using an available set of statistics S.

Chaudhuri: Col. 6, lines 48-60

Query optimizer 224 may use any suitable statistics of any suitable structure for query optimization. A statistic is a summary structure associated with a set of one or more columns in a relation. One commonly used statistical descriptor is a histogram. Database server 220 may store statistics in system catalog tables 226, for example.

A set of statistics S can be denoted by a set comprising single-columns and/or multicolumns. Thus, the set {R.sub.1.a, R.sub.1.c, (R.sub.2.c, R.sub.2.d)} represents a set of three statistics comprising single-column statistics on R.sub.1.a, that is on column a of relation R.sub.1 ; and R.sub.1.c and also comprising multi-column statistics on the two-column combination (R.sub.2.c, R.sub.2.d). The notation (R.sub.2.c, R.sub.2.d) denotes a two-dimensional statistic on columns c and d of relation R.sub.2. The number of statistics in the set S is denoted by .vertline.S.vertline..

Other pertinent portions of Chaudhuri are set forth below:

Chaudhuri: Col. 1, lines 33-54

Statistics may be created and maintained on a table, an index, a single column of a table, or combinations of columns of a table, although the structure of statistics may vary from system to system. Single column statistics typically comprise a histogram of values in the domain of that column and may include one or more of the following parameters: the number of distinct values in the column, the density of values in the column, and the second highest and the

second lowest values in the column. Multi-column statistics typically represent information on the distribution of values over the Cartesian product of the domains in it. As one example, multi-column statistics on (R.sub.2.c, R.sub.2.d) may contain information on the joint distribution of values over R.sub.2.c and R.sub.2.d. In Microsoft.RTM. SQL Server, for example, such multi-column statistics would contain joint density information and a histogram on the leading dimension R.sub.2.c. The single and multi-column statistics available for a database make cost estimation significantly more accurate and help the query optimizer arrive at better query execution plans. In the absence of statistics, cost estimates can be dramatically different often resulting in a poor choice of the execution plan.

The above portions of Chaudhuri describe the use of statistics by a query optimizer in choosing among query execution plans for a query. These statistics used by the query optimizer, however, are created and maintained on a table, an index, a single column of a table, or combinations of columns of a table. Specifically, in generating execution plans, the query optimizer of Chaudhuri relies on statistics on columns of tables referenced in a query to estimate the cost in time to execute the query using more than one possible execution plan for the query, and then chooses among the possible execution plans for the query.

Unlike Appellant's invention, however, Chaudhuri says nothing about the use of statistics on expressions of pre-defined queries to determine an optimal query execution plan for a query. Indeed, nothing in the above portions of Chaudhuri can fairly be said to represent the same limitations as Appellant's independent claims 1, 12 and 33.

Consequently, the Chaudhuri reference does not teach or suggest all of the limitations of Appellant's claimed invention. Moreover, the various elements of Appellant's claimed invention together provide operational advantages over the Chaudhuri reference. In addition, Appellant's invention solves problems not recognized by the Chaudhuri reference.

Thus, Appellant's attorney submits that independent claims 1, 12 and 23 are allowable over Chaudhuri. Further, dependent claims 3-10, 13-20 and 23-30 are submitted to be allowable over Chaudhuri in the same manner, because they are dependent on independent claims 1, 11 and 21, respectively, and because they contain all the limitations of the independent claims.

2. Claims 2, 13 and 24

With regard to dependent claims 2, 13, and 24, which recite that each of the pre-defined queries is associated with an automatic summary table, a materialized view or a view, these claims stand or fall with independent claims 1, 12 and 23.

VIII. CONCLUSION

In light of the above arguments, Appellant's attorney respectfully submits that the cited references do not anticipate nor render obvious the claimed invention. More specifically, Appellant's claims recite novel physical features which patentably distinguish over any and all references under 35 U.S.C. §§ 102 and 103.

As a result, a decision by the Board of Patent Appeals and Interferences reversing the Examiner and directing allowance of the pending claims in the subject application is respectfully solicited.

Respectfully submitted,

GATES & COOPER LLP
Attorneys for Appellant

Howard Hughes Center
6701 Center Drive West, Suite 1050
Los Angeles, California 90045
(310) 641-8797

Date: January 17, 2011

By: /George H. Gates/
Name: George H. Gates
Reg. No.: 33,500

GHG/

CLAIMS APPENDIX

1. (PREVIOUSLY PRESENTED) A method of optimizing execution of a query that accesses data stored on a data store connected to a computer, comprising:

using statistics on one or more expressions of one or more pre-defined queries to determine an optimal query execution plan for the query; and

executing the optimal query execution plan for the query in order to access the data stored on the data store connected to a computer and then output the accessed data.

2. (ORIGINAL) The method of claim 1, wherein each of the pre-defined queries is associated with an automatic summary table, a materialized view or a view.

3. (ORIGINAL) The method of claim 1, further comprising:

generating cardinality estimates for one or more query execution plans for the query using the statistics of one or more of the pre-defined queries that vertically overlap the query; and

using the generated cardinality estimates to determine an optimal query execution plan for the query.

4. (ORIGINAL) The method of claim 3, wherein the statistics are used to improve a combined selectivity estimate of one or more predicates of the query.

5. (ORIGINAL) The method of claim 4, wherein the predicates are applied by one or more of the pre-defined queries.

6. (ORIGINAL) The method of claim 5, wherein the selectivity estimate comprises a ratio of a cardinality of the pre-defined query to a product of cardinalities of base tables referenced in the pre-defined query and the query.

7. (ORIGINAL) The method of claim 4, wherein zero or more predicates of the query are applied by one of the pre-defined queries and wherein the remaining predicates are eligible to be applied on the pre-defined query.

8. (ORIGINAL) The method of claim 7, wherein a predicate is eligible to be applied on the pre-defined query if it can be evaluated using the output columns and expressions of the pre-defined query.

9. (ORIGINAL) The method of claim 8, further comprising determining a subpredicate combined selectivity estimate of the unapplied eligible predicates using column distribution statistics of the pre-defined query.

10. (ORIGINAL) The method of claim 9, wherein a cardinality ratio comprises a ratio of a cardinality of the pre-defined query to a product of cardinalities of base tables referenced in the pre-defined query and the query.

11. (ORIGINAL) The method of claim 10, wherein the selectivity estimate comprises a product of the subpredicate combined selectivity estimate and the cardinality ratio.

12. (PREVIOUSLY PRESENTED) An apparatus for optimizing execution of a query, comprising:

a computer having a data store coupled thereto, wherein the data store stores data;

one or more computer programs, performed by the computer, for using statistics on one or more expressions of one or more pre-defined queries to determine an optimal query execution plan for the query, and executing the optimal query execution plan for the query in order to access the data stored on the data store connected to a computer and then output the accessed data.

13. (ORIGINAL) The apparatus of claim 12, wherein each of the pre-defined queries is associated with an automatic summary table, a materialized view or a view.

14. (ORIGINAL) The apparatus of claim 12, further comprising:

one or more computer programs for generating cardinality estimates for one or more query execution plans for the query using the statistics of one or more of the pre-defined queries that vertically overlap the query; and

one or more computer programs for using the generated cardinality estimates to determine an optimal query execution plan for the query.

15. (ORIGINAL) The apparatus of claim 14, wherein the statistics are used to improve a combined selectivity estimate of one or more predicates of the query.

16. (ORIGINAL) The apparatus of claim 15, wherein the predicates are applied by one or more of the pre-defined queries.

17. (ORIGINAL) The apparatus of claim 16, wherein the selectivity estimate comprises a ratio of a cardinality of the pre-defined query to a product of cardinalities of base tables referenced in the pre-defined query and the query.

18. (ORIGINAL) The apparatus of claim 15, wherein zero or more predicates of the query are applied by one of the pre-defined queries and wherein the remaining predicates are eligible to be applied on the pre-defined query.

19. (ORIGINAL) The apparatus of claim 18, wherein a predicate is eligible to be applied on the pre-defined query if it can be evaluated using the output columns and expressions of the pre-defined query.

20. (ORIGINAL) The apparatus of claim 19, further comprising one or more computer programs for determining a subpredicate combined selectivity estimate of the unapplied eligible predicates using column distribution statistics of the pre-defined query.

21. (ORIGINAL) The apparatus of claim 20, wherein a cardinality ratio comprises a ratio of a cardinality of the pre-defined query to a product of cardinalities of base tables referenced in the pre-defined query and the query.

22. (ORIGINAL) The apparatus of claim 21, wherein the selectivity estimate comprises a product of the subpredicate combined selectivity estimate and the cardinality ratio.

23. (PREVIOUSLY PRESENTED) An article of manufacture comprising a program storage medium readable by a computer and embodying one or more instructions executable by the computer to optimizing execution of a query that accesses data stored on a data store connected to the computer, comprising:

using statistics on one or more expressions of one or more pre-defined queries to determine an optimal query execution plan for the query; and

executing the optimal query execution plan for the query in order to access the data stored on the data store connected to a computer and then output the accessed data.

24. (ORIGINAL) The article of claim 23, wherein each of the pre-defined queries is associated with an automatic summary table, a materialized view or a view.

25. (ORIGINAL) The article of claim 23, further comprising:

generating cardinality estimates for one or more query execution plans for the query using the statistics of one or more of the pre-defined queries that vertically overlap the query; and

using the generated cardinality estimates to determine an optimal query execution plan for the query.

26. (ORIGINAL) The article of claim 25, wherein the statistics are used to improve a combined selectivity estimate of one or more predicates of the query.

27. (ORIGINAL) The article of claim 26, wherein the predicates are applied by one or more of the pre-defined queries.

28. (ORIGINAL) The article of claim 27, wherein the selectivity estimate comprises a ratio of a cardinality of the pre-defined query to a product of cardinalities of base tables referenced in the pre-defined query and the query.

29. (ORIGINAL) The article of claim 26, wherein zero or more predicates of the query are applied by one of the pre-defined queries and wherein the remaining predicates are eligible to be applied on the pre-defined query.

30. (ORIGINAL) The article of claim 29, wherein a predicate is eligible to be applied on the pre-defined query if it can be evaluated using the output columns and expressions of the pre-defined query.

31. (ORIGINAL) The article of claim 30, further comprising determining a subpredicate combined selectivity estimate of the unapplied eligible predicates using column distribution statistics of the pre-defined query.

32. (ORIGINAL) The article of claim 31, wherein a cardinality ratio comprises a ratio of a cardinality of the pre-defined query to a product of cardinalities of base tables referenced in the pre-defined query and the query.

33. (ORIGINAL) The article of claim 32, wherein the selectivity estimate comprises a product of the subpredicate combined selectivity estimate and the cardinality ratio.

EVIDENCE APPENDIX

None.

RELATED PROCEEDINGS APPENDIX

A Board Decision was rendered on July 19, 2010 for a related appeal filed in U.S. Patent Application Serial No. 09/669,556, which is the parent of this application. A copy of that Board Decision is attached hereto.

A Board Decision was rendered on May 20. 2005 for a related appeal filed in U.S. Patent No. 7,080,062. A copy of that Board Decision is attached hereto.